

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Matic Stare

Napovedovanje možnih trkov med vlaki in predori

MAGISTRSKO DELO

MAGISTRSKI ŠTUDIJSKI PROGRAM DRUGE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA
SMER: RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: doc. dr. Uroš Čibej

SOMENTOR: /

Ljubljana, 2025

To delo je ponujeno pod licenco *Creative Commons Priznanje avtorstva-Deljenje pod enakimi pogoji 2.5 Slovenija* (ali novejšo različico). To pomeni, da se tako besedilo, slike, grafi in druge sestavine dela kot tudi rezultati zaključnega dela lahko prosto distribuirajo, reproducirajo, uporabljajo, priobčujejo javnosti in predelujejo, pod pogojem, da se jasno in vidno navede avtorja in naslov tega dela in da se v primeru spremembe, preoblikovanja ali uporabe tega dela v svojem delu, lahko distribuira predelava le pod licenco, ki je enaka tej. Podrobnosti licence so dostopne na spletni strani creativecommons.si ali na Inštitutu za intelektualno lastnino, Streliška 1, 1000 Ljubljana.



Izvorna koda zaključnega dela, njeni rezultati in v ta namen razvita programska oprema je ponujena pod licenco GNU General Public License, različica 3 (ali novejša). To pomeni, da se lahko prosto distribuira in/ali predeluje pod njenimi pogoji. Podrobnosti licence so dostopne na spletni strani <http://www.gnu.org/licenses/>.

ZAHVALA

Na tem mestu zapišite, komu se zahvaljujete za izdelavo magistrske naloge. V zahvali se poleg mentorja spodobi omeniti vse, ki so s svojo pomočjo prispevali k nastanku vašega izdelka.

Matic Stare, 2025

Vsem rožicam tega sveta.

*"The only reason for time is so that
everything doesn't happen at once."*

— Albert Einstein

Kazalo

Povzetek

Abstract

1	Uvod	1
1.1	Opis problema	1
1.2	Motivacija in cilji dela	1
1.3	Prispevki magistrske naloge	2
1.4	Struktura magistrske naloge	3
2	Pregled sorodnih del	5
2.1	Zaznavanje trkov v prostoru	5
2.2	Obdelava oblakov točk	6
2.3	Metode prostorskega indeksiranja	6
2.4	Analitično modeliranje v železniškem prometu	6
2.5	Prehodne krivulje v železniškem prometu	7
2.6	Primerjava obstoječih pristopov	7
3	Teoretične osnove	9
3.1	Geometrijska predstavitev predorov	9
3.2	Geometrijski model vagona	10
3.2.1	Pozicioniranje z medosno razdaljo	10
3.2.2	Ortogonalni koordinatni sistem	11
3.2.3	Simulacija premika vagona	11

KAZALO

3.2.4	Kritične točke za računanje trkov	12
3.3	B-zlepki (B-splines)	13
3.3.1	Matematične osnove B-zlepkov	13
3.3.2	Interpolacija in aproksimacija	13
3.4	Metode za merjenje razdalj in določanje strani	14
3.4.1	Razdalja do parametrične krivulje	14
3.4.2	Določanje strani	14
4	Metodologija in pristop	15
4.1	Pregled predlaganega pristopa	15
4.2	Predobdelava vhodnih podatkov	16
4.2.1	Obdelava oblakov točk predora	16
4.2.2	Modeliranje vagona	16
4.2.3	Obdelava kontrolnih točk	16
4.3	Generiranje kontrolnih točk	16
4.4	Modeliranje gibanja vagona	16
4.4.1	Izračun pozicije na osnovi medosne razdalje	16
4.4.2	Določitev kritičnih točk vagona	16
4.4.3	Izračun krivulj gibanja	16
4.5	Algoritmi za zaznavanje trkov	16
4.5.1	Varnostni odmik	16
4.5.2	Algoritem zaznavanja	17
5	Implementacija	19
5.1	Arhitektura sistema	19
5.2	Ključni moduli implementacije	19
5.2.1	TunnelSlicer – obdelava geometrije predora	19
5.2.2	TrainGenerator – modeliranje vlaka	19
5.2.3	CollisionDetector – zaznavanje trkov	19
5.2.4	Simulation – simulacija gibanja	19
5.3	B-zlepki za stene predora	19
5.4	Transformacije koordinatnih sistemov	19

KAZALO

5.5	Simulacija gibanja vagona	19
5.6	Vizualizacija simulatorja	19
6	Eksperimentalno ovrednotenje	21
6.1	Testni scenariji in podatki	21
6.1.1	Predor Ringo	21
6.1.2	Predor Globoko	21
6.2	Parametri sistema	21
6.3	Evalvacijski kriteriji	21
6.4	Rezultati testiranja	22
6.4.1	Natančnost zaznavanja trkov	22
6.4.2	Računska učinkovitost	22
6.4.3	Analiza varnostnih razdalj	22
6.5	Primerjava z obstoječimi metodami	22
6.6	Diskusija rezultatov	22
7	Sklepne ugotovitve	23
7.1	Povzetek prispevkov	23
7.2	Omejitve pristopa	23
7.3	Predlogi za nadaljnje delo	23

Seznam uporabljenih kratic

kratica	angleško	slovensko
CA	classification accuracy	klasifikacijska točnost
DBMS	database management system	sistem za upravljanje podatkovnih baz
SVM	support vector machine	metoda podpornih vektorjev
...

Povzetek

Naslov: Napovedovanje možnih trkov med vlaki in predori

V vzorcu je predstavljen postopek priprave magistrskega dela z uporabo okolja L^AT_EX. Vaš povzetek mora sicer vsebovati približno 100 besed, ta tukaj je odločno prekratek. Dober povzetek vključuje: (1) kratek opis obravnavanega problema, (2) kratek opis vašega pristopa za reševanje tega problema in (3) (najbolj uspešen) rezultat ali prispevek magistrske naloge.

Ključne besede

železniški promet, zaznavanje trkov, analitično modeliranje, oblaki točk

Abstract

Title: Predicting Possible Collisions Between Trains and Tunnels

This sample document presents an approach to typesetting your BSc thesis using L^AT_EX. A proper abstract should contain around 100 words which makes this one way too short. A good abstract contains: (1) a short description of the tackled problem, (2) a short description of your approach to solving the problem, and (3) (the most successful) result or contribution in your thesis.

Keywords

rail transport, collision detection, analytical modeling, point clouds

Poglavje 1

Uvod

1.1 Opis problema

V železniškem prometu je zagotavljanje varnosti v predorih ključnega pomena, še posebej pri dolgi in široki tovarni kompoziciji, ki se giblje skozi ozke in ukrivljene predore. Problem, ki ga obravnavam v tej magistrski nalogi, je zaznavanje morebitnih trkov med vlakom in stenami predora, ki nastanejo zaradi nepravilnega sledenja predpisanemu varnostnemu prostoru ali napak v modeliranju geometrije predora.

Klasične metode, kot je uporaba minimalnega prereza, so v takšnih scenarijih nezadostne, saj ne upoštevajo kompleksne ukrivljenosti poti ali relativnih premikov vagona, ki lahko presežejo varnostne meje, zlasti v ostrih ovinkih. Problem je izrazit pri dolgi tovarni kompoziciji, kjer razlika med položajem sprednje in zadnje osi povečuje tveganje za trk. Poleg tega trenutne metode pogosto niso dovolj prilagodljive za različne geometrije predorov in vlakov.

1.2 Motivacija in cilji dela

Motivacija za delo izhaja iz realnega izziva, ki sem ga prejel od podjetja Slovenske železnice. Ti so izrazili potrebo po razvoju avtomatiziranega sistema,

ki bi omogočil natančno zaznavanje trkov med vlakom in predorom.

V tej nalogi predlagam pristop, ki temelji na obdelavi oblaka točk predora in analitičnem modeliranju gibanja vlaka. Osnovna vhodna podatka sta oblak točk predora, pridobljen s 3D laserskim skenerjem, in kontrolne točke, ki definirajo pot železniške proge. Na podlagi teh podatkov sistem obdela geometrijo predora, generira B-zlepke za stene predora v različnih horizontalnih plasteh ter simulira gibanje vagona vzdolž kontrolnih točk. Med simulacijo se izvaja zaznavanje trkov s preverjanjem razdalj med kritičnimi točkami vagona in stenami predora.

Naloga se umešča na področje računalniškega modeliranja in analize v prostoru ter prinaša novost v kombinaciji obdelave oblakov točk s simulacijo gibanja vlaka in zaznavanjem trkov v realnem času.

1.3 Prispevki magistrske naloge

Magistrska naloga bo prispevala k razvoju sistema za zaznavanje trkov med vlakom in predorom s simulacijo gibanja vlaka. V primerjavi z obstoječimi metodami, ki temeljijo na statični analizi minimalnih prerezov, predlagana rešitev omogoča dinamično simulacijo gibanja in kontinuirano preverjanje varnostnih razdalj.

Novost naloge je v integraciji obdelave oblakov točk predora z analitičnim modeliranjem gibanja vagona vzdolž ukrivljene poti ter implementaciji sistema za zaznavanje trkov v realnem času. Glavni prispevki magistrske naloge so:

- Razvoj sistema za obdelavo oblakov točk predora z B-zlepki za reprezentacijo sten
- Implementacija simulacije gibanja vagona vzdolž kontrolnih točk z ortogonalnim koordinatnim sistemom
- Sistem za zaznavanje trkov z analizo razdalj med kritičnimi točkami vagona in stenami predora

- Praktična aplikacija za Slovenske železnice z možnostjo nadaljnega razvoja

1.4 Struktura magistrske naloge

Poglavje 2

Pregled sorodnih del

2.1 Zaznavanje trkov v prostoru

Na področju zaznavanja trkov v prostoru se pogosto uporabljajo metode, ki temeljijo na analizi oblakov točk in algoritmih prostorskega indeksiranja. Ena izmed najpogostejše uporabljenih tehnik je uporaba k-d dreves za učinkovito iskanje sosednjih točk v prostoru, kot je prikazano v delu Schauerja in Nüchterja [1]. Prednost njihovega pristopa je visoka računska učinkovitost pri analizi oblakov točk velikega obsega. Ker pa je točk zelo veliko, se poraja potreba po bolj pametnih izračunih trkov. Njihov članek bo služil kot osnova za to magistrsko delo.

Kot alternativo klasičnim metodam so Hermann et al. [2] razvili algoritme, ki temeljijo na vokselizaciji prostora. Ti algoritmi omogočajo hitro preverjanje prostorske zasedenosti, vendar lahko pri zelo natančnih analizah izgubijo detajle zaradi diskretizacije prostora.

V delu Niwa in Masuda [3] je predstavljen pristop za zaznavanje trkov z metodo globinskih slik, kar izboljša učinkovitost in pravilnost. Ta pristop omogoča zanesljivejše zaznavanje trkov v gostih oblakih točk, vendar ima še vedno veliko časovno in prostorsko zahtevnost.

2.2 Obdelava oblakov točk

Klein in Zachmann [4] obravnavata zaznavanje trkov s pomočjo implicitnih površin, ustvarjenih iz oblakov točk. Njihov pristop je posebej uporaben pri obdelavi kompleksnih geometrij, vendar je računsko zahteven, kar lahko omejuje uporabo v realnem času.

Avtorji Li et al. [5] pregledajo najnovejše pristope strojnega učenja za obdelavo LiDAR podatkov. Izpostavljajo, kako lahko globoko učenje izboljša zaznavanje in analizo oblakov točk v avtonomnih vozilih, še posebej pri neenakomernih in šumnih podatkih. Kljub napredku se metode soočajo z izzivi pri obdelavi velikih oblakov točk in zagotavljanjem rezultatov v realnem času, kar omejuje njihovo uporabnost v hitro spreminjajočih se okoljih.

2.3 Metode prostorskega indeksiranja

Prostorsko indeksiranje je ključno za učinkovito obdelavo velikih oblakov točk. K-d drevesa, kot jih uporabljajo Schauer in Nüchter [1], omogočajo hitro iskanje najbližjih sosedov v večdimenzionalnih prostorih. Te strukture podatkov so posebej primerne za aplikacije, kjer je potrebno pogosto iskanje točk v določeni okolici.

Vendar pa tradicionalne metode prostorskega indeksiranja pogosto niso optimalne za dinamične scenarije, kjer se objekti gibljejo skozi prostor. V takšnih primerih je potreben pristop, ki upošteva časovno komponento gibanja.

2.4 Analitično modeliranje v železniškem prometu

Everett et al. [6] predstavijo sistem za izogibanje trkom v dinamičnih okoljih z uporabo globokega spodbujevalnega učenja. Prednost tega pristopa je prilagodljivost za različne scenarije in obdelava spremenljivega števila agen-

tov brez strogih predpostavk o njihovem gibanju. Kljub temu metoda manj poudarja analizo geometrijskih lastnosti, kar jo omejuje pri natančnih prostorskih analizah, kot je analiza trkov med vlakom in predorom, zaradi česar je njena uporaba v tem kontekstu manj primerna.

2.5 Prehodne krivulje v železniškem prometu

V železniškem prometu so prehodne krivulje ključne za zagotavljanje gladkega prehoda med ravnimi in ukrivljenimi odseki prog. Brustad in Dalmo [7] analizirajo prehodne krivulje, ki omogočajo gladek prehod med ravnimi in ukrivljenimi odseki železniških tirov. Glavna prednost teh krivulj je njihova sposobnost zmanjšanja sil in obrabe vozil ter tirnic, kar povečuje udobje potnikov in zmanjšuje stroške vzdrževanja. Kljub temu se raziskave na tem področju še vedno soočajo z izzivi, kot so določanje optimalnih lastnosti krivulj za različne scenarije in vozne profile.

Jiang et al. [8] predlagajo uporabo paraboličnih in sinusoidnih prehodnih krivulj za zmanjšanje dolgovalovnih nepravilnosti v vertikalnih profilih tirov. Prednost tega pristopa je zmanjšanje pospeškov pri prehodih, kar izboljša stabilnost vlaka in varnost potnikov. Slabost pa je, da metoda zahteva precizno načrtovanje in prilagoditev specifičnim konstrukcijskim zahtevam, kar lahko poveča začetne stroške implementacije.

2.6 Primerjava obstoječih pristopov

Iz zgoraj predstavljenih del je razvidno, da večina obstoječih metod bodisi zanemara dinamične lastnosti gibanja bodisi ne omogoča učinkovitega prilagajanja različnim geometrijam. Metode, ki temeljijo na obdelavi oblakov točk [1, 3], so računsko zahtevne in pogosto niso primerne za analizo v realnem času. Po drugi strani pristopi strojnega učenja [5, 6] omogočajo prilagodljivost, vendar ne zagotavljajo teoretično podprtih rezultatov, ki so potrebni za varnostno kritične aplikacije v železniškem prometu.

Cilj te magistrske naloge je preseči omejitve obstoječih pristopov z vključitvijo analitičnega modeliranja gibanja kritičnih točk in prekrivanjem teh krivulj z geometrijo predora, kar bo omogočilo natančnejše in hitrejše zaznavanje trkov.

Poglavje 3

Teoretične osnove

V tem poglavju predstavimo teoretične osnove, ki so potrebne za razumevanje predlaganega pristopa k zaznavanju trkov med vlakom in predorom. Osredotočimo se na geometrijsko predstavitev objektov v prostoru, matematične osnove B-zlepkov ter algoritme za zaznavanje trkov.

3.1 Geometrijska predstavitev predorov

Predori so predstavljeni z oblaki točk, pridobljenih s 3D laserskim skenerjem. Oblak točk predstavlja diskretno vzorčenje površine predora, kjer vsaka točka $\mathbf{p}_i = (x_i, y_i, z_i)$ določa prostorsko pozicijo na steni predora. Oblak točk $\mathcal{P} = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_n\}$ vsebuje n točk, ki skupaj opisujejo geometrijo predora.

V našem pristopu obravnavamo podatke predorov, ki so organizirani kot zaporedni prečni prerezi vzdolž osi predora. Vsak presek je definiran z množico 2D koordinat (x, y) , ki opisujejo obris predora na določeni poziciji vzdolž osi z .

Glavni izziv pri obdelavi takšnih podatkov je njihova transformacija v uporaben 3D oblak točk, ki ohranja geometrijske lastnosti predora. To dosežemo z naslednjim pristopom:

- **Postavitev oblaka točk v 3D:** 2D koordinate (x, y) za vsak presek dopolnimo z z koordinato, ki ustreza poziciji vzdolž poti

- **Transformacija koordinatnega sistema:** podatki se transformirajo iz globalnega koordinatnega sistema v lokalni koordinatni sistem predora, kjer z -os sovpada s središčno linijo predora

Matematična formulacija transformacije je podana z:

$$\mathbf{p}'_i = \mathbf{R}(\mathbf{p}_i - \mathbf{t}_{\text{center}}) \quad (3.1)$$

kjer je \mathbf{R} rotacijska matrika, izpeljana iz tangentnega vektorja središčne linije, $\mathbf{t}_{\text{center}}$ pa translacijski vektor do središča predora.

Za konstrukcijo rotacijske matrike \mathbf{R} uporabimo Rodriguesovo rotacijsko formulo [9]:

$$\mathbf{R} = \mathbf{I} + \sin(\theta)\mathbf{K} + (1 - \cos(\theta))\mathbf{K}^2 \quad (3.2)$$

kjer je θ kot med vektorjema, \mathbf{K} pa antisimetrična matrika osi rotacije. Namen te transformacije je prilagoditi oblak točk dejanskemu poteku predora v prostoru. Ker so prvotni podatki organizirani kot ravni prerezi, rotacijska transformacija zagotovi, da se oblak točk čim bolj prilega dejanski ukrivljeni geometriji predora. To je ključno za natančno predstavitev prostorskih odnosov med vlakom in stenami predora ter posledično za zanesljivo zaznavanje trkov.

3.2 Geometrijski model vagona

Vagon je modeliran kot pravokotno telo (kvader) z dimenzijami širina w , višina h in globina d . Geometrijska predstavitev temelji na konceptu medosne razdalje in ortogonalnem koordinatnem sistemu.

3.2.1 Pozicioniranje z medosno razdaljo

Pozicija vagona je določena s točkama \mathbf{p}_0 in \mathbf{p}_1 , ki sta oddaljeni za medosno razdaljo l_{wb} . Za določitev sprednje osi uporabljamo metodo presečišča kroga in krivulje.

Sprednja os \mathbf{p}_1 se nahaja na presečišču kroga s središčem v \mathbf{p}_0 in radijem l_{wb} z B-zlepke kontrolnih točk:

$$\|\mathbf{C}(t) - \mathbf{p}_0\|^2 = l_{wb}^2 \quad (3.3)$$

kjer je $\mathbf{C}(t)$ parametrična predstavitev B-zlepke kontrolnih točk in $t \in [0, 1]$ parameter krivulje.

Za rešitev te enačbe iščemo ničle funkcije:

$$f(t) = \|\mathbf{C}(t) - \mathbf{p}_0\|^2 - l_{wb}^2 \quad (3.4)$$

Ta pristop zagotavlja, da je medosna razdalja natančno ohranjena tudi na ukrivljenih odsekih poti, kar je ključno za realistično modeliranje gibanja vagona.

3.2.2 Ortogonalni koordinatni sistem

Orientacija vagona v prostoru je določena z ortogonalnim koordinatnim sistemom:

$$\mathbf{f} = \frac{\mathbf{p}_1 - \mathbf{p}_0}{\|\mathbf{p}_1 - \mathbf{p}_0\|} \quad (\text{naprej}) \quad (3.5)$$

$$\mathbf{u} = (0, 1, 0) \quad (\text{navzgor}) \quad (3.6)$$

$$\mathbf{r} = \mathbf{u} \times \mathbf{f} \quad (\text{desno}) \quad (3.7)$$

3.2.3 Simulacija premika vagona

Za potrebe simulacije gibanja vagona je potrebno določiti celoten geometrijski model, ki omogoča natančno predstavitev vagona v prostoru. Vagon je modeliran z osmimi vogali pravokotnega telesa, ki se izračunajo na podlagi dejanskih mej vagona.

Najprej definirajmo sprednjo in zadnjo mejo vagona:

$$\mathbf{p}_{\text{rear}} = \mathbf{p}_0 - \mathbf{f} \cdot (d \cdot \omega) \quad (3.8)$$

$$\mathbf{p}_{\text{front}} = \mathbf{p}_0 + \mathbf{f} \cdot (d \cdot (1 - \omega)) \quad (3.9)$$

kjer je ω parameter odmika koles, d globina vagona in \mathbf{f} enotski vektor v smeri naprej.

Osem vogalov vagona je nato definiranih kot:

$$\mathbf{v}_{i,j,k} = \mathbf{p}_{\text{base}} + i \cdot \frac{w}{2} \mathbf{r} + j \cdot h \cdot \mathbf{u} + k \cdot \frac{d}{2} \cdot \mathbf{f} \quad (3.10)$$

kjer so $i, k \in \{-1, 1\}$, $j \in \{0, 1\}$ in $\mathbf{p}_{\text{base}} = \frac{\mathbf{p}_{\text{rear}} + \mathbf{p}_{\text{front}}}{2}$ je središče med zadnjo in sprednjo mejo vagona.

Ta pristop omogoča popolno geometrijsko predstavitev vagona v prostoru ter je ključen za vizualizacijo in analizo gibanja celotne strukture vagona vzdolž ukrivljene poti.

3.2.4 Kritične točke za računanje trkov

Za učinkovito zaznavanje trkov se na vsaki višinski ravnini y v predoru uporablja optimiziran nabor šestih kritičnih točk, ki predstavljajo najkritičnejše pozicije za možne kolizije s stenami. Te točke so razporejene v treh vzdolžnih pozicijah (zadaj, sredina, spredaj) na obeh stranskih robovih vagona.

Kritične točke za zaznavanje kolizij na višini y so definirane s formulo:

$$\mathbf{c}_{s,p} = \mathbf{p}_{\text{base},p} + s \cdot \frac{w}{2} \mathbf{r} + y \cdot \mathbf{u} \quad (3.11)$$

kjer je:

$$s \in \{-1, 1\} \quad (\text{levo/desno})$$

$$p \in \{\text{back, middle, front}\} \quad (\text{zadaj, sredina, spredaj})$$

$$\mathbf{p}_{\text{base,back}} = \mathbf{p}_{\text{rear}}$$

$$\mathbf{p}_{\text{base,middle}} = \frac{\mathbf{p}_{\text{rear}} + \mathbf{p}_{\text{front}}}{2}$$

$$\mathbf{p}_{\text{base,front}} = \mathbf{p}_{\text{front}}$$

kjer je \mathbf{r} enotski vektor v desno smer, \mathbf{u} enotski vektor navzgor in w širina vagona.

Ta pristop generira za vsako horizontalno ravnino predora (definirano z y koordinato) natanko šest kritičnih točk, ki pokrivajo celotno širino in dolžino vagona. Sistematična razporeditev omogoča zanesljivo zaznavanje kršitev varnostnih razdalj ali situacij, kjer bi se vagon nahajal zunaj dovoljenih mej predora.

3.3 B-zlepki (B-splines)

B-zlepki so parametrične krivulje, ki omogočajo gladko interpolacijo ali aproksimacijo množice točk. V našem sistemu jih uporabljamo za reprezentacijo sten predora ter za modeliranje poti vagona.

3.3.1 Matematične osnove B-zlepkov

B-zlepek stopnje p [10] je definiran s kontrolnimi točkami $\mathbf{C}_0, \mathbf{C}_1, \dots, \mathbf{C}_n$ in vozliščnim vektorjem $\mathbf{t} = \{t_0, t_1, \dots, t_{n+p+1}\}$:

$$\mathbf{S}(y) = \sum_{i=0}^n \mathbf{C}_i B_{i,p}(y) \quad (3.12)$$

kjer so $B_{i,p}(y)$ B-zlepek bazne funkcije, definirane rekurzivno:

$$B_{i,0}(y) = \begin{cases} 1 & \text{če } t_i \leq y < t_{i+1} \\ 0 & \text{sicer} \end{cases} \quad (3.13)$$

$$B_{i,p}(y) = \frac{y - t_i}{t_{i+p} - t_i} B_{i,p-1}(y) + \frac{t_{i+p+1} - y}{t_{i+p+1} - t_{i+1}} B_{i+1,p-1}(y). \quad (3.14)$$

3.3.2 Interpolacija in aproksimacija

Pri obdelavi oblakov točk predora uporabljamo dva pristopa:

Interpolacija: B-zlepek natančno prehaja skozi vse podane točke. To je ustrezno, ko želimo ohraniti vse detajle geometrije predora.

Aproksimacija: B-zlepek se prilagodi splošni obliki podatkov, vendar ne prehaja natančno skozi vse točke. To je koristno za zmanjšanje vpliva šuma v meritvah.

3.4 Metode za merjenje razdalj in določanje strani

Za zaznavanje trkov je ključno merjenje razdalj med točkami vagona in stenami predora ter določanje, na kateri strani B-zlepka se točka nahaja.

3.4.1 Razdalja do parametrične krivulje

Razdalja med točko \mathbf{q} in B-zlepkom $\mathbf{C}(u)$ je definirana kot:

$$d(\mathbf{q}, \mathbf{C}) = \min_{u \in [0,1]} \|\mathbf{q} - \mathbf{C}(u)\| \quad (3.15)$$

To je optimizacijski problem, ki ga rešujemo numerično z iskanjem najbližje točke na krivulji.

3.4.2 Določanje strani

Za določitev, ali se točka nahaja znotraj ali zunaj predora, uporabljamo predznak razdalje. Za 2D preseku predora uporabimo vektorski produkt za določitev strani:

$$s = \text{sign}(\mathbf{t} \times (\mathbf{q} - \mathbf{C}(u^*))) \quad (3.16)$$

kjer pozitiven predznak pomeni, da je točka na levi strani krivulje, negativen pa na desni.

Poglavje 4

Metodologija in pristop

4.1 Pregled predlaganega pristopa

Pri magistrski nalogi se osredotočam na pristop k zaznavanju trkov med vlakom in predorom, ki vključuje več korakov. Kot vhodni podatki se uporabljajo oblaki točk predora, pridobljeni s 3D laserskim skenerjem, ter kontrolne točke, ki definirajo pot železniške proge.

Metodologija vključuje obdelavo oblakov točk predora s transformacijo vzdolž ukrivljene poti, generiranje horizontalnih prerezov predora z B-zlepki za reprezentacijo sten, modeliranje vagona kot kvadra ter simulacijo gibanja vagona vzdolž kontrolnih točk. Med simulacijo se izvaja zaznavanje trkov s preverjanjem razdalj med kritičnimi točkami vagona in stenami predora z določenim varnostnim odmikom.

Sistem je implementiran v programskem jeziku Python z uporabo knjižnic PyVista za vizualizacijo, NumPy za numerične izračune in SciPy za interpolacijo z B-zlepkami.

4.2 Predobdelava vhodnih podatkov

4.2.1 Obdelava oblakov točk predora

4.2.2 Modeliranje vagona

4.2.3 Obdelava kontrolnih točk

4.3 Generiranje kontrolnih točk

4.4 Modeliranje gibanja vagona

4.4.1 Izračun pozicije na osnovi medosne razdalje

4.4.2 Določitev kritičnih točk vagona

4.4.3 Izračun krivulj gibanja

4.5 Algoritmi za zaznavanje trkov

Zaznavanje trkov temelji na preverjanju dveh pogojev: ali se kritične točke vagona nahajajo preblizu stenami predora (kršitev varnostnega odmika) ali zunaj predora.

4.5.1 Varnostni odmik

Varnostni odmik δ predstavlja minimalno dovoljeno razdaljo med vagom in steno predora. Kršitev nastane, ko:

$$d(\mathbf{v}_i, \mathbf{C}_{\text{stena}}) < \delta \quad (4.1)$$

kjer je \mathbf{v}_i kritična točka vagona in $\mathbf{C}_{\text{stena}}$ B-zlepek stene predora.

4.5.2 Algoritem zaznavanja

Glavni algoritem zaznavanja trkov deluje v naslednjih korakih:

1. Za vsako kritično točko vagona v_i
2. Izračunaj razdaljo do leve stene d_L in desne stene d_R
3. Določi, na kateri strani sten se točka nahaja
4. Preverti kršitve:
 - Če $d_L < \delta$ ali $d_R < \delta$: kršitev varnostnega odmika
 - Če je točka na napačni strani: zunaj predora

Psevdokoda algoritma je predstavljena v Algoritmu 1.

Algorithm 1 Zaznavanje trkov med vagom in predorom

```

1: violations  $\leftarrow \emptyset$ 
2: for vsako kritično točko  $v_i$  vagona do
3:    $d_L \leftarrow$  razdalja od  $v_i$  do leve stene
4:    $d_R \leftarrow$  razdalja od  $v_i$  do desne stene
5:    $s_L \leftarrow$  stran glede na levo steno
6:    $s_R \leftarrow$  stran glede na desno steno
7:   if  $d_L < \delta$  then
8:     dodaj kršitev: "preblizu levi steni"
9:   end if
10:  if  $d_R < \delta$  then
11:    dodaj kršitev: "preblizu desni steni"
12:  end if
13:  if  $s_L < 0$  ali  $s_R > 0$  then
14:    dodaj kršitev: "zunaj predora"
15:  end if
16: end for
17: return violations
  
```

Poglavje 5

Implementacija

5.1 Arhitektura sistema

5.2 Ključni moduli implementacije

5.2.1 TunnelSlicer – obdelava geometrije predora

5.2.2 TrainGenerator – modeliranje vlaka

5.2.3 CollisionDetector – zaznavanje trkov

5.2.4 Simulation – simulacija gibanja

5.3 B-zlepki za stene predora

5.4 Transformacije koordinatnih sistemov

5.5 Simulacija gibanja vagona

5.6 Vizualizacija simulatorja

Poglavje 6

Eksperimentalno ovrednotenje

6.1 Testni scenariji in podatki

6.1.1 Predor Ringo

6.1.2 Predor Globoko

6.2 Parametri sistema

6.3 Evalvacijski kriteriji

Evalvacija sistema je bila izvedena z analizo delovanja na dveh testnih scenarijih. Preverjalo se je pravilno zaznavanje kršitev varnostnih razdalj, stabilnost sistema med simulacijo ter ustreznost vizualizacije rezultatov. Sistem je uspešno zaznal situacije, kjer se vagon približa preblizu stenam predora ali presega dovoljene meje predora.

6.4 Rezultati testiranja

6.4.1 Natančnost zaznavanja trkov

6.4.2 Računska učinkovitost

6.4.3 Analiza varnostnih razdalj

6.5 Primerjava z obstoječimi metodami

6.6 Diskusija rezultatov

Poglavje 7

Sklepne ugotovitve

7.1 Povzetek prispevkov

7.2 Omejitve pristopa

7.3 Predlogi za nadaljnje delo

Literatura

- [1] J. Schauer, A. Nüchter, Efficient point cloud collision detection and analysis in a tunnel environment using kinematic laser scanning and k-d tree search, *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* XL-3 (2014) 289 – 295.
URL <https://doi.org/10.5194/isprsarchives-XL-3-289-2014>
- [2] A. Hermann, F. Drews, J. Bauer, S. Klemm, A. Roennau, R. Dillmann, Unified gpu voxel collision detection for mobile manipulation planning, *IEEE International Conference on Intelligent Robots and Systems* (09 2014). doi:10.1109/IRoS.2014.6943148.
- [3] T. Niwa, H. Masuda, Interactive collision detection for engineering plants based on large-scale point-clouds, *Computer-Aided Design and Applications* 13 (4) (2016) 511–518. arXiv:<https://doi.org/10.1080/16864360.2015.1131546>, doi:10.1080/16864360.2015.1131546.
URL <https://doi.org/10.1080/16864360.2015.1131546>
- [4] J. Klein, G. Zachmann, Point Cloud Surfaces using Geometric Proximity Graphs, *Computers and Graphics* 28 (6) (2004).
- [5] Y. Li, L. Ma, Z. Zhong, F. Liu, D. Cao, J. Li, M. A. Chapman, Deep learning for lidar point clouds in autonomous driving: A review, *arXiv preprint arXiv:2005.09830* (2020). arXiv:2005.09830.
URL <https://arxiv.org/abs/2005.09830>

-
- [6] M. Everett, Y. F. Chen, J. P. How, Collision avoidance in pedestrian-rich environments with deep reinforcement learning, *IEEE Access* 9 (2021) 10357–10377. doi:10.1109/access.2021.3050338.
URL <http://dx.doi.org/10.1109/ACCESS.2021.3050338>
- [7] T. F. Brustad, R. Dalmo, Railway transition curves: A review of the state-of-the-art and future research, *Infrastructures* 5 (5) (2020). doi:10.3390/infrastructures5050043.
URL <https://www.mdpi.com/2412-3811/5/5/43>
- [8] L. Jiang, Y. Li, Y. Zhao, M. Cen, The characteristics of long-wave irregularities in high-speed railway vertical curves and method for mitigation, *Sensors* 24 (13) (2024). doi:10.3390/s24134403.
URL <https://www.mdpi.com/1424-8220/24/13/4403>
- [9] J. S. Dai, Euler–rodriques formula variations, quaternion conjugation and intrinsic connections, *Mechanism and Machine Theory* 92 (2015) 144–152. doi:<https://doi.org/10.1016/j.mechmachtheory.2015.03.004>.
URL <https://www.sciencedirect.com/science/article/pii/S0094114X15000415>
- [10] M. S. Hasan, M. N. Alam, M. Fayz-Al-Asad, N. Muhammad, C. Tunç, B-spline curve theory: An overview and applications in real life, *Nonlinear Engineering* 13 (1) (2024) 20240054 [cited 2025-08-16]. doi:doi:10.1515/nleng-2024-0054.
URL <https://doi.org/10.1515/nleng-2024-0054>